

УДК 519.163, 65.011.56

Дубинин И. С., Арапов С. Ю., Тягунов А. Г.

УрФУ, г. Екатеринбург, Россия

МЕТОД ГЕНЕРАЦИИ СОЧЕТАНИЙ ДЛЯ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ

Аннотация

В предыдущих работах решалась задача составления оптимального производственного расписания. Для оптимизации процесса потребовался метод генерации сочетаний, пригодный для использования на графических процессорах. Обзор литературы показал, что предлагаемые решения не оптимальны, поэтому было решено разработать такой метод самостоятельно. В основе разработанного метода лежит другой принцип представления комбинаторных объектов. Благодаря этому номера объектов, составляющих выборку, можно вычислять алгебраически, без использования циклов и операций ветвления.

Ключевые слова: алгоритм, сочетания, выборка, комбинаторная оптимизация, параллельные вычисления.

Dubinina I. S., Arapov S. Yu., Tjagunov A. G.

UrFU, Ekaterinburg, Russia

METHOD OF GENERATING COMBINATIONS FOR PARALLEL COMPUTING

Abstract

In previous works we solved the problem of optimal production schedules. To optimize the process, a method of generating combinations, suitable for use on graphics processing units, was needed. Literature review showed that the proposed solutions are not optimal, so it was decided to develop a method of their own. The basis of this method is another principle of representation of combinatorial objects. This allows computing algebraically sample's objects numbers, without using loops and branching operations.

Keywords: algorithm, combination, combinatorial optimization, parallel computing.

Введение

Для небольших полиграфических производств характерен большой поток коротких заказов, длительность выполнения которых не превышает одну смену. Так как многие заказы отличаются параметрами печати и используемыми материалами, между печатью двух заказов необходим процесс переналадки оборудования. В случае с небольшими заказами суммарное время переналадок может занимать значительную долю от продолжительности рабочей смены. В таких условиях составление производственного расписания является важной, но трудоемкой задачей. Для упрощения и ускорения данного процесса используются разнообразные автоматизированные системы управления.

В статьях [1–3] описывался процесс разработки АСУ, учитывающей специфику полиграфической отрасли. В ней для составления расписания использовался метод локального поиска. Выбор метода обусловлен его высокой эффективностью для данного класса задач [4]. Для формирования локальной окрестности был использован метод [5; 6]. Однако в данном методе широко используются операции ветвления, что не только уменьшает производительность, но и делает невозможным его использование на *SIMD*-процессорах, в частности на процессорах видеокарт [7].

Целью данной работы является создание метода генерации сочетаний, то есть получения номеров объектов из индекса нумерации, не использующего циклы и операции ветвления, а также пригодного для дальнейшего использования на процессорах архитектуры *SIMD*.

1. Решение

На рис. 1 приведено графическое представление сочетаний по 2 из 5 и 6. Строки соответствуют первому элементу сочетания, столбцы — второму. Так как выбор элемента для сочетания возможен только один раз, столбцы соответствуют элементам со сдвигом в единицу. Например, индекс 8 (рис. 1, а) находится в третьей строке и третьем столбце. Это расшифровывается следующим образом: данному объекту соответствует выбор третьего (третья строка) и четвертого (третий столбец) элементов исходного массива.

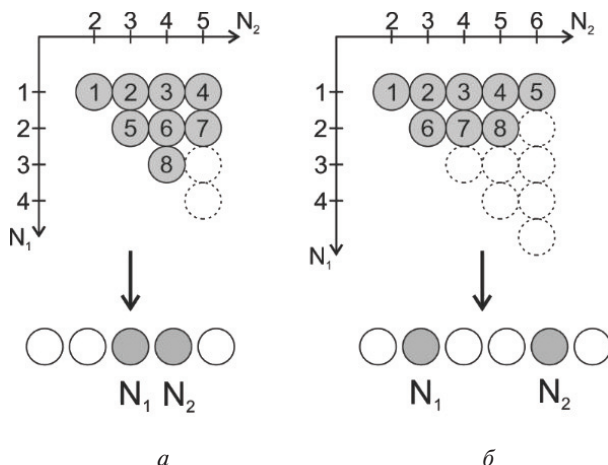


Рис. 1. Графическое представление выборов:
а) по 2 из 5; б) по 2 из 6

Проблема традиционного подхода нумерации объектов заключается в том, что для определения номеров элементов необходимо знать общее их количество. На рис. 1 видно, что сочетания под одинаковыми индексами из разного количества объектов подразумевают очевидно разные объекты.

Чтобы избежать данной особенности, предлагается использовать нумерацию «с другой стороны». На рис. 2 изображено графическое представление предлагаемого порядка перечисления сочетаний из конечного набора элементов по три. Предложенная методика хороша тем, что при любом количестве элементов в исходном наборе индекс нумерации всегда будет строго соответствовать сочетанию.

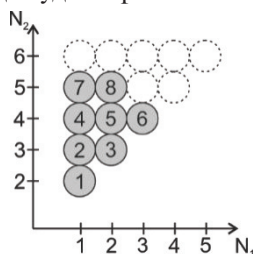


Рис. 2. Сочетания по 2 из 6 с использованием предложенного алфавитно-графического порядка

Для лучшего понимания принципа работы можно провести геометрическую аналогию. Так, каждому сочетанию будет соответствовать шарик, а множество всех возможных сочетаний можно представить в виде n -мерной пирамиды основанием вверх (рис. 3). По мере нарастания индекса нумерации пирамида будет заполняться шариками начиная с вершины. В этом случае, для нахождения сочетания, соответствующего индексу интересующего нас шарика, нужно будет последовательно вычислять ее координаты. Начало координат находится в вершине пирамиды.

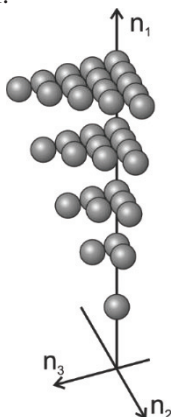


Рис. 3. Графическое представление сочетаний трех объектов. Трем координатам каждого шарика соответствуют три заказа.

Таким образом, индекс сочетания по три определяется формулой:

$$I = \frac{n_1(n_1 + 1)(n_1 + 2)}{6} + \frac{n_2(n_2 + 1)}{2} + n_3, \quad (1)$$

где

$$n_1 = N_1 - 2, n_2 = N_2 - 1, n_3 = N_3. \quad (2)$$

Решать уравнение (1) следует в следующем порядке. Сначала необходимо решить уравнение

$$x_1(x_1 + 1)(x_1 + 2) = 6I \quad (3)$$

и округлить полученный результат в меньшую сторону до ближайшего целого. Полученное значение и будет являться основой n_1 для получения индекса сочетания N_1 с помощью выражений (2):

$$n_1 = \text{fix}(x_1), \text{ или } n_1 = \text{floor}(x_1). \quad (4)$$

Далее необходимо найти разность

$$I_1 = I - n_1 \quad (5)$$

и решить следующее уравнение:

$$x_2(x_2 + 1) = 2I_1, \quad (6)$$

откуда аналогичным образом получаем n_2 :

$$n_2 = \text{fix}(x_2), \text{ или } n_2 = \text{floor}(x_2). \quad (7)$$

Разность

$$n_3 = I_1 - n_2 \quad (8)$$

даст последнее недостающее значение n_3 . После этого с помощью выражений (2) могут быть получены выборочные индексы.

Решение уравнения (6) не представляет никакой сложности:

$$x_2 = \frac{\sqrt{I_1 + 1} - 1}{2}, \quad (9)$$

второй корень квадратного уравнения (6) необходимо исключить.

Решение кубического уравнения (3) несколько сложнее предыдущего:

$$x_1 = \sqrt[3]{3I + \sqrt[3]{9I^2 - 1/27}} + \frac{1}{\sqrt[3]{3I + \sqrt[3]{9I^2 - 1/27}}} - 1, \quad (10)$$

это единственный положительный корень, остальные являются комплексными числами и могут быть исключены из дальнейшего рассмотрения.

2. Результат исследования

Проверка эффективности описанного алгоритма проводилась путем его применения к программе составления производственного расписания, написанной на языке *MATLAB*. Для сравнения использовалась программа, использующая принцип работы, описанный в [5].

Исследование производительности осуществлялось на компьютере с двумя процессорами *Intel Xeon X5670* (12 вычислительных потоков). Тестирование производилось на одинаковом наборе исходных данных. Было осуществлено по 100 запусков каждой из программ, для сравнения было взято среднее арифметическое времени выполнения.

Программе с описанным алгоритмом для нахождения решения требуется 9,6 с, тогда как программа, использующая классический метод нумерации выполняется за 13 с. Таким образом, прирост производительности составил примерно 35 %.

3. Выводы

В ходе работы был получен метод генерации сочетаний, превосходящий по быстродействию используемый ранее. Тестирование программы, использующей данный метод, показало его эффективность. Путем оптимизации вспомогательных алгоритмов было получено заметное ускорение работы АСУ. На практике это позволяет увеличить количество обрабатываемых заказов либо снизить системные требования программы.

Список литературы

1. Некоторые допустимые приближения в задаче составления оптимальных производственных расписаний для полиграфических предприятий / С. Ю. Арапов [и др.] // Известия высших учебных заведений. Проблемы полиграфии и издательского дела. 2010. № 4. С. 89–95.
2. Применение параллельных вычислений к решению задачи оптимизации производственных расписаний для полиграфического предприятия / Арапов С. Ю. [и др.] // Известия высших учебных заведений. Проблемы полиграфии и издательского дела. 2010. № 5. С. 82–90.
3. Дубинин И. С., Арапов С. Ю., Тягунов А. Г. Решение задачи поиска оптимального порядка работ с учетом затрат на переналадку оборудования // Международная научно-практическая конференция «СВЯЗЬ-ПРОМ 2011», в рамках VI международного форума «СВЯЗЬ-ПРОМЭКСПО 2011», Екатеринбург. Сборник трудов.
4. Lin S., Kernighan B. W. An Effective Heuristic Algorithm for the Traveling-Salesman Problem // Operations Research. 1973. № 2. Vol. 21. P. 498–516.
5. Тимошевская Н. Е. О нумерации перестановок и сочетаний для организации параллельных вычислений в задачах проектирования управляющих систем // Известия Томского политехнического университета. 2004. Том 307. № 6. С. 18–20.
6. Тимошевская Н. Е. Разработка и исследование параллельных комбинаторных алгоритмов // Прикладная дискретная математика. 2009. № 2 (4). С. 96–103.

7. Оценка эффективности работы алгоритма локального поиска при оптимизации расписания в одностадийной системе / С. Ю. Арапов, Д. А. Тарасов, Ю. Н. Колмогоров, И. С. Дубинин // Международная научно-практическая конференция «СВЯЗЬ-ПРОМ 2009», в рамках VI международного форума «СВЯЗЬ-ПРОМЭКСПО 2009», посвященного 150-летию со дня рождения изобретателя радио А. С. Попова: материалы междунар. науч.-практ. конф. (Екатеринбург, 17–19 марта 2009 г.) / отв. ред. Е. А. Субботин. Екатеринбург: УрТИСИ ГОУ ВПО «СибГУТИ», 2009.